Journal of Physics: Conference Series

Towards a global monitoring system for CMS computing operations

--Manuscript Draft--

Manuscript Number:	
Full Title:	Towards a global monitoring system for CMS computing operations
Article Type:	Contributed
Corresponding Author:	Andrea Sciabà, Ph.D. CERN Genève, SWITZERLAND
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	CERN
Corresponding Author's Secondary Institution:	
First Author:	Andrea Sciabà, Ph.D.
First Author Secondary Information:	
Order of Authors:	Andrea Sciabà, Ph.D.
	Lothar Bauerdick
Order of Authors Secondary Information:	

Towards a global monitoring system for CMS computing operations

L A T Bauerdick[1] **and A Sciabà**[2]

[1] Fermi National Accelerator Laboratory, PO Box 500, Batavia IL, USA

[2] European Organization for Nuclear Research, CH-1211 Genève 23, Switzerland

E-mail: bauerdick@fnal.gov Andrea.Sciaba@cern.ch

Abstract. The operation of the CMS computing system requires a complex monitoring system to cover all its aspects: central services, databases, the distributed computing infrastructure, production and analysis workflows, the global overview of the CMS computing activities and the related historical information. Several tools are available to provide this information, developed both inside and outside of the collaboration and often used in common with other experiments. Despite the fact that the current monitoring allowed CMS to successfully perform its computing operations, an evolution of the system is clearly required, to adapt to the recent changes in the data and workload management tools and models and to address some shortcomings that make its usage less than optimal. Therefore, a recent and ongoing coordinated effort was started in CMS, aiming at improving the entire monitoring system by identifying its weaknesses and the new requirements from the stakeholders, rationalise and streamline existing components and drive future software development. This contribution gives a complete overview of the CMS monitoring system and a description of all the recent activities that have been started with the goal of providing a more integrated, modern and functional global monitoring system for computing operations.

1. Introduction

The CMS computing system is operating since 2008, much before the start of LHC proton-proton data taking. Its role is to enable the collaboration to process, store, distribute and analyse data collected by the CMS experiment. As for the other LHC experiments, it is built over the infrastructure and the services provided by the Worldwide LHC Computing Grid (WLCG) as an upper layer of services and tools that take into account the CMS data model and implement the CMS computing model.

The operation of such complex system requires substantial effort from CMS, from WLCG and the sites; it is therefore of utmost importance to be able to know in detail the system status and its history at all times, as the availability of good monitoring information is a prerequisite for efficient operations. In November 2010 CMS critically reviewed all the available monitoring with the goal of identifying its weaknesses and finding ways to improve it.

The purpose of this paper is *i)* to give an overview of the CMS monitoring system, the areas covered by it, the existing tools and the known deficiencies of the system, *ii)* to describe the process pursued by CMS to achieve a global monitoring system based on a limited number of common tools and *iii)* to talk about the ongoing developments along the lines defined by the global strategy.

2. Monitoring areas

The CMS offline and computing monitoring can be divided into several areas, with very different scope and functionality. They are:

- *Service monitoring.* The monitoring directly provided by each service, which describes the service status and health and is primarily addressed to service operators.
- *Facility monitoring.* It describes the status of the distributed computing infrastructure in terms of used and available resources and their health via automated functional tests; it is addressed both to computing operators and sites.
- *Global overview.* This type of monitoring describes the current status of the computing activities across the whole infrastructure; it is addressed to activity coordinators.
- *Historical accounting.* The facility and activity monitoring information, when permanently archived, allows to show trends, for example in activity levels, resource usage and health which are essential for planning and resource optimisation. The target of this information is very ample and includes computing coordinators.

In the rest of this section we describe in more detail the existing monitoring tools for each area.

2.1. Service monitoring

A fair amount of effort in CMS was devoted to provide computing operators with a complete view of the status of the CMS computing services. This is done at three levels:

- at CERN, the nodes on which CMS services run are monitored via Lemon to collect and display relevant metrics and generate alarms [1];
- the service has its own native monitoring to provide both operators and users of the service with all the information they need to know if the service is healthy and operational. Examples are the PhEDEx monitoring [2], the CRAB server monitoring and the WMAgent monitoring [3];
- a global overview of the CMS computing services is provided via SLS, the monitoring information aggregation and visualisation tool developed and used by the CERN IT for the computing centre [4]. The status of computing services as measured by SLS is further aggregated in the Critical Services map developed by the Dashboard team [5].

Any CMS service is required to provide at least a heartbeat monitor to know the status of its components, though the implementation may vary significantly: for example, PhEDEx looks for any agent activity by querying its central database (used for all information exchange), while Frontier relies on probing every few minutes each CMS Squid server and on the status of the Tomcat server in the Frontier front-ends.

Some services also provide information about their internal state. For example, CRAB servers give access to component log files, the status of the component message queues or the users submitting to them. On top of that, monitoring pages exist to aggregate information coming from different CRAB servers.

Service monitoring shows the highest level of heterogeneity, for obvious reasons most of it is strongly coupled with the service itself and different services are based on different software technologies and languages. For service operators, who are concerned only with one service, this is not really an issue, but it is not the case for people doing computing shifts, who need to check several services and are much more interested in having a uniform presentation of the service information. This is the main motivation behind using the Critical Services map and SLS.

SLS requires a single mandatory metric (called *availability* and defined to be between 0 and 100), but it is highly customizable for what concerns complementary information. CMS has recently invested some effort in the improvement of the SLS information [6]. SLS is able to generate notifications when the availability drops below certain configurable thresholds.

Although the quality of service monitoring is not uniformly good across services, it is generally sufficient for their correct operation. However, a feature sorely missed by computing operations is the possibility to generate and manage alarm conditions in a uniform way: currently alarms are generated by Lemon, SLS and via custom notifications by some services but are not centrally collected (with the exception soon of WMAgent – as will be explained later). A general alarm management system would be extremely desirable and could be used not only by others in CMS (for example, the online group) but also by other experiments.

2.2. Facility monitoring

This area addresses everything related to the status of the distributed infrastructure whether the Grid interfaces and the site services used by CMS are working, the occupancy of computing and storage resources and the functionality of the relevant network links. The main monitoring sources are the Service Availability Monitor (SAM) [7], HammerCloud [8], PhEDEx, the WLCG Information System and the batch systems of the Tier-1 sites. The aggregation and visualisation layer is provided by several tools, as the CMS Site Readiness [9], the Site Status Board [10], the Service Usability Monitor [11] and HappyFace [12].

The Service Availability Monitor is used since 2006 by CMS to run custom functional tests on computing and storage services to test several aspects of the services themselves and of the CMS setup at the sites. In particular, the site availability (defined as the fraction of time a site passed all tests) has been for long the main quality metric, until it was superseded by the more comprehensive Site Readiness metric, which combines it with other metrics, like the success rate of test jobs and of test transfers over the relevant network links.

The Site Readiness was highly successful as a system to measure the usability of a site and it is being progressively integrated into the Site Status Board, which will make easy for other LHC experiments to adopt it; its future will in any case take into account the recommendations of the WLCG Technical Evolution Groups to achieve an even higher degree of standardization and simplification, possibly extending the use of SAM as a framework to publish test results from external sources.

A minor – but useful – project was carried out to provide a uniform representation of the batch system information from the Tier-1 sites to address the needs of the computing operations team. Batch system-specific plugins are used to format the relevant data as standard XML files and a dedicated HappyFace instance. HappyFace is an aggregation and visualisation tool for monitoring information widely used in German WLCG sites and was particularly suited for this application (altough in principle any visualisation tool capable of reading the XML feeds could be used). Currently, there are no plans to extend the project to Tier-2 sites because of the amount of effort needed to have the XML generators deployed everywhere, but it could be easily adopted by other experiments.

Visualisation of the SAM functional tests was recently moved from an old, CMS-only application to a new interface, the Service Usability Monitor, used by all the LHC experiments. An alternative is represented by MyWLCG [13], developed by the SAM team and directly supported by WLCG, which, if adopted, would be yet another step towards a uniform approach to site functional testing.

Another important ingredient in the CMS facility monitoring is the ability to run test jobs that realistically simulate analysis or production jobs and use their success rate as a quality metric. In this way it is possible to test the global system, from data discovery to job submission, job execution, read access to storage and output retrieval. Very recently HammerCloud superseded the CMS Job Robot, so CMS now uses the same tool as ATLAS and LHCb. Still, further work is needed to achieve a more realistic representation of the site behaviour, in particular by replacing the job submission via the EMI WMS with the submission via glideinWMS [14]; in fact, only a fraction of the analysis activity still relies on the EMI WMS

Table 1. Sources of global overview monitoring.

Source	Monitored activities
T0Mon	Tier-0 activities
Global Monitor	WMAgent workflows
CMS Dashboard	Analysis task and job information
PhEDEx monitoring	Data transfers (at the dataset level)
FTS monitoring	Data transfers (at the file level)
WLCG Transfers Dashboard	Data transfers (at the file level) (in development)
Storage accounting	Disk usage (in development)

and all production is done via glideins.

2.3. Global Overview

Under this category we include all monitoring that shows “what is happening now” (or has recently happened) in terms of activities: user analysis jobs, production workflows, data transfers; as such, its main target is represented by production operators and end users from the collaboration. There are several sources and pages available, the main being listed in table 1 and shortly described below.

All the Tier-0 workflows are monitored by T0Mon. It works as a web interface querying the Tier-0 Data Service, a REST interface to the Tier-0 Activity State Tracker (T0AST) [15]. As it only works with the old production system, it will be decommissioned at the same time by the end of 2012. After that, its function will be performed by the WMAgent native monitoring, the Global Monitor.

Production workflows running outside the Tier-0 are covered by the Global Monitor, developed as part of the WMAgent; it allows to see the details of the workflow definition, the status of the workflow jobs and provides access to the job outputs. It is a relatively simple web interface built over the WMAgent REST interface, giving direct access to the internal WMAgent databases.

The Dashboard Historical View is used by production teams also to monitor the current status and the recent past of several aggregated quantities, like the number of running and terminated jobs by site and by activity, their success rate and CPU efficiency. It is used by both ATLAS and CMS, its plots can be easily embedded by other web applications or widgets and the information exported in several standard formats (XML, JSON, etc.).

End users submitting analysis jobs are served by the CMS Dashboard applications, in particular the Interactive View and the Task Monitoring [5]. Both were developed by the CERN IT, first for CMS and then ported to ATLAS. The Interactive View allows to access a plethora of information about all CMS jobs, limited to the latest seven days to ensure scalability, while the Task Monitoring is the primary monitoring page for users.

Information about transfers comes from two different sources: the PhEDEx monitoring and the FTS monitoring developed by CMS and IN2P3 [16]. The former is used to monitor dataset movement on replication and the latter uses information extracted from the database of an FTS instance. The FTS monitoring is a powerful tool for configuration optimisation and performance studies, as it collects very complete statistics on FTS transfers. More recently a WLCG Global Trasfer Monitor was developed with the aim of providing a far greater level of functionality and it is being validated [17].

Table 2. Quantities for historical accounting.

Description
LHC duty cycle
Rate per primary dataset
Number of pile-up interactions
Processing time per event for prompt reconstruction
Size per event for RAW, RECO and AOD data
Number of events per primary dataset for RAW, RECO and AOD data
Memory usage for processing jobs
Latency for prompt reconstruction, skimming jobs and archival al Tier-1 sites
CPU efficiency per site and activity
Job success rate per site and activity
Resource usage versus pledged resources per site
Running jobs per site and activity
Tape usage per data type and site
Data volume written per day
Transfer efficiencies and rates
Transfer latency

Currently, storage accounting is almost exclusively based on information extracted from PhEDEx; for example, a web page shows the distribution of registered data by type and custodiality status and by site. Nonetheless, there is an ongoing development aimed at collecting detailed information about the usage of storage resources by CMS [18]. For each site it will allow to know how much disk space is used by data tier, physics group, data type (Monte Carlo, real data) and user via a web interface and an API. This will fill a gap between the information obtainable from PhEDEx (but only for data registered in the CMS data bookkeeping system) and the information obtainable from the WLCG information system (only grand totals of used and available space). Information from the storage is conveniently generated by *storage dumps* according to a format agreed in WLCG among the LHC experiments and the main storage technology developers (CASTOR, dCache, DPM); this makes the CMS storage accounting in principle usable by any virtual organisation.

The lack of a proper integration of all these monitoring sources is a known limitation and is currently replaced by the use of custom web pages embedding external plots or in the worst case just links. As we will see later, having a framework capable of integrating diverse monitoring sources has been identified as a critical need for the CMS monitoring.

2.4. Historical accounting

The availability of historical records for the monitored systems is often very useful and in some cases absolutely essential. Many of the monitoring systems used in CMS provide historical records. By historical accounting however we refer to information relevant to reconstruct long term usage trends. A long list of quantities to be tracked for computing operations was identified and it is summarized in table 2.

The tools used currently (or in future) to display historical accounting information are listed in table 3; most of them were listed also in the previous section and will not be rediscussed here.

The development of an accounting portal is one of the most awaited improvements in CMS monitoring. In its first instance it will use the CMS Overview framework to collect and organize

Table 3. Sources of historical accounting.

Source	Monitored activity
PhEDEx monitoring	Data transfers
Storage accounting	Disk usage (in development)
CMS Dashboard	Historical information on CMS jobs
Accounting portal	Historical accounting information (in development)

the required information that is already readily available from existing sources. A second step would be to add the possibility of showing correlations between different quantities, but a study to choose a technology for cross correlation of heterogeneous data sources has yet to be started.

3. The CMS Monitoring Task Force

The review of the CMS monitoring on November 2010 concluded that work was needed to improve the overall coordination of the monitoring efforts and the interaction between the operations groups and the development groups. The panel issued some recommendations, that can be summarized as follows:

- define a formal process for interaction between different groups of monitoring developers;
- centralization of the collection of use cases, prioritisation of new requirements and definition of a work plan by appointing a monitoring responsible;
- aim at developing a coherent monitoring overview of CMS systems and operations with a single entry point;
- ensure that all job monitoring information is reliably sent to the CMS Dashboard by the workload management infrastructure;
- CMS Dashboard developers should give more priority to consolidation and performance than to new features.

All recommendations were accepted by the collaboration and the task of their implementation was assigned to a Monitoring Task Force (MTF), with an expected duration of nine months. The MTF had two coordinators and several members, including contacts for every offline and computing project and software developers from the CMS collaboration and the CERN IT department.

Given that one important source of inefficiency in monitoring efforts was the lack of a formal process for collecting requirements, the MTF started that process as one of its first tasks. Requirements were defined according to a common format and rated according to their importance. This initial phase was essential to prepare for the monitoring workshop held in May 2011, during which all requirements were described by the project stakeholders and the current state of the art and the ongoing developments were illustrated.

The workshop's main goal was to define a realistic workplan for the duration of the MTF activity and beyond. The main tasks of the workplan are listed in table 4.

For each item, the MTF identified a group of people responsible for its execution. While some items were relatively straightforward and could be completed in a short time, others spawned development projects that are still ongoing. Finally, a few of them will require a long term project and therefore fall well beyond the MTF lifetime and are hence topics of discussion at the coordination level.

Table 4. Monitoring Task Force workplan.

Description
Draw the overall architectural picture of the CMS monitoring
Select an aggregation technology to create custom views of monitoring information
Define requirements and identify a suitable solution for generating and aggregating alarms
Consolidate and clean up information stored in the Dashboard
Define performance goals for the Dashboard and improve its performance
Build a portal for accounting information
Implement a full accounting for the storage utilisation
Consolidate plans about data transfer issues
Identify the information needed for application performance and validation studies
Bring into operations the Data Popularity service

The status of the workplan to date is rather positive. Architectural diagrams for most of the monitoring systems have been collected; a technology to create custom monitoring views was identified; requirements for an alarm aggregation system were defined; many issues with the information reporting to the Dashboard from the WMAgent were finally solved and the performance of the CMS Dashboard improved in some cases by 1-2 orders of magnitude (mostly thanks to database-level and query optimisations); data transfer monitoring is now converging on two tools, PhEDEx and an official WLCG system; metrics relevant for application performance were identified and are now collected by HammerCloud; finally, the Data Popularity service is now in production [19].

4. Development projects

The most significant monitoring development projects are:

- the CMS Overview framework
- the CMS accounting portal
- the WMAgent Alert Framework
- the CMS Storage Accounting
- the Data Popularity service

The CMS Overview framework was selected by an independent reviewer over HappyFace and the Dashboard as the basis of a unified presentation layer of the CMS monitoring system. Such layer would be used to provide “big picture” views from the same web site instead of having to refer to multiple, heterogeneous web sites. This approach was very successful in the CMS Data Quality Monitor and indeed the Overview framework shares much of its source code with it.

Overview does not have any persistent storage. It has a backend, written in Python and using Cherrypy as server framework, that obtains the needed data via specialized plugins and transforms them into the desired way, and a frontend, a Javascript modular web application using YUI as a GUI library and protovis/matplotlib for the plotting. The backend retrieves (and caches) the data whenever the frontend requests it.

The rationale behind the choice of Overview is twofold: on one side, it is based on the assumption that every monitoring component must be the only authoritative source of its data (including historical information); on another side, having full control over the development and the deployment of a visualisation framework that is already fully integrated in the CMS infrastructure gives obvious advantages.

The CMS Overview is the basis for two ongoing developments: the CMS accounting portal mentioned above and the new web page used by computing shifts people. For those, implementation is still in an early phase and the goal is to have as soon as possible usable prototypes and add new features incrementally.

The Alert framework is a system to centrally collect and publish alert-worthy events from WMAgent instances, to the benefit of the service operators [20]; its functionality overlaps with the requirements for an alarm aggregator system collected in the MTF, and it is already foreseen to be applied also to other systems (for example PhEDEx). It is still to be decided whether to further expand the scope of the system.

The Data Popularity service is a project born from the need to optimise data placement and storage space allocation at Tier-2 sites by measuring how much the existing data is used at sites in terms of number of accesses and users, CPU hours of processing and successes versus failures [19]. It was recently extended to directly collect information from XRootD servers [21] and it is now used in computing operations.

5. Conclusions

One year after the Monitoring Task Force was established as a temporary coordination body for monitoring, significant progress has been made and most of its goals have been reached. In particular, it was very successful in facilitating discussions and braimstorming among all the interested parties, in improving the awareness on the importance of monitoring and in setting a direction for current and future developments.

In particular, two trends were strongly encouraged: the usage of systems and tools not specific to CMS (notably SAM/Nagios, HammerCloud, Dashboard applications and other) and the consolidation into one (or more realistically, a few) entry points to all monitoring information. More recently, CMS was closely involved in the WLCG Technical Evolution Group on Operations and Tools for what concerns infrastructure and network testing and it is planned to continue working along the recommendations that WLCG will eventually issue on how to evolve monitoring in the next years.

The MTF itself has completed its planned time span but there is vast agreement about the need for a long term coordination activity for monitoring, though it was not yet decided in what form. What is clear is that much still needs to be done to achieve all the remaining goals before the end of the long LHC shutdown and that even further synergies with other experiments and CERN should be sought.

Acknowledgments

We would like to thank all members of the CMS Monitoring Task Force and the people and the institutes that contributed to its activity and achievements.

References

[1] Cancio G *et al* 2005 *Proc. Computing in High-Energy Physics (CHEP'04)* CERN-2005-002, ed A Aimar *et al* (Geneva: CERN) pp 1145–48
[2] Egeland R *et al* 2010 *J. Phys.: Conf. Ser.* **219** 062010
[3] Wakefield S 2012 The CMS workload management system *J. Phys.: Conf. Ser.* (not yet published)
[4] Lopienski S 2008 *J. Phys.: Conf. Ser.* **119** 052025
[5] Saiz P *et al* 2012 Experiment Dashboard - a generic, scalable solution for monitoring of the LHC computing activities, distributed sites and services *J. Phys.: Conf. Ser.* (not yet published)
[6] Molina-Pérez J A *et al* 2012 Monitoring techniques and alarm procedures for CMS services and sites in WLCG *J. Phys.: Conf. Ser.* (not yet published)
[7] Collados D *et al* 2010 *J. Phys.: Conf. Ser.* **219** 062008

[8] Van Der Ster D C *et al* 2012 Experience in Grid site testing for ATLAS, CMS and LHCb with HammerCloud *J. Phys.: Conf. Ser.* (not yet published)

[9] Flix J, Hernández J and Sciabà A 2011 *J. Phys.: Conf. Ser.* **331** 072020

[10] Saiz P *et al* 2012 Collaborative development. Case study of the development of flexible monitoring applications *J. Phys.: Conf. Ser.* (not yet published)

[11] Sciabà A *et al* 2012 New solutions for large scale functional tests in the WLCG infrastructure with SAM/Nagios: the experiments experience *J. Phys.: Conf. Ser.* (not yet published)

[12] Büge V *et al* 2010 *J. Phys.: Conf. Ser.* **219** 062057

[13] Lapka W *et al* 2012 Distributed monitoring infrastructure for Worldwide LHC Computing Grid *J. Phys.: Conf. Ser.* (not yet published)

[14] Sfiligoi I *et al* 2010 The pilot way to Grid resources using glideinWMS *Preprint* FERMILAB-CONF-09-373-CD

[15] Hufnagel D 2011 *J. Phys.: Conf. Ser.* **331** 032017

[16] Flix J *et al* 2011 *J. Phys.: Conf. Ser.* **331** 052013

[17] Andreeva J *et al* 2012 Providing WLCG Global Transfer monitoring *J. Phys.: Conf. Ser.* (not yet published)

[18] Ratnikova N *et al* 2012 Data storage accounting and verification in LHC experiments *J. Phys.: Conf. Ser.* (not yet published)

[19] Giordano D *et al* 2012 Implementing data placement strategies for the CMS experiment based on a popularity model *J. Phys.: Conf. Ser.* (not yet published)

[20] Maxa Z 2012 Alert Messaging in the CMS Distributed Workload System *J. Phys.: Conf. Ser.* (not yet published)

[21] Tadel M *et al* 2012 XRootD Monitoring for the CMS experiment *J. Phys.: Conf. Ser.* (not yet published)